



Doctoral Thesis

Simulation of neural networks on parallel computers

Author(s):

Müller, Urs A.

Publication Date:

1993

Permanent Link:

<https://doi.org/10.3929/ethz-a-000896274> →

Rights / License:

[In Copyright - Non-Commercial Use Permitted](#) →

This page was generated automatically upon download from the [ETH Zurich Research Collection](#). For more information please consult the [Terms of use](#).

Diss. ETH No 10188

Simulation of Neural Networks on Parallel Computers

A dissertation submitted to the
SWISS FEDERAL INSTITUTE OF TECHNOLOGY
ZURICH

for the degree of
Doctor of Technical Sciences

presented by

Urs A. Müller
Dipl. El.-Ing. ETH
born September 3, 1963
citizen of Unterkulm AG

accepted on the recommendation of
Prof. Dr. W. Guggenbühl, examiner
Prof. Dr. F. Eggimann, co-examiner

1993

Zusammenfassung

In den letzten Jahren sind künstliche neuronale Netze immer beliebter geworden. Im Unterschied zu klassischen Systemen werden sie nicht programmiert, sondern anhand von Beispielen trainiert und können deshalb auch bei Problemen eingesetzt werden, von denen keine exakt beschreibbaren Lösungswege bekannt sind. Wegen der benötigten Flexibilität sind für Forschungsarbeiten Simulationen auf digitalen Computern am besten geeignet. Die dazu notwendige Rechenleistung ist jedoch immer noch ein limitierender Faktor. Diese Situation kann durch den Einsatz von Parallelrechnern entscheidend verbessert werden.

Die vorliegende Dissertation enthält Beiträge zur Simulation neuronaler Netze auf Parallelrechnern. Für die praktische Umsetzung der erarbeiteten Konzepte ist das am Institut entwickelte MUSIC System (Multiprocessor System with Intelligent Communication) verwendet worden. In einem ersten Schritt wurde der bekannte "Back-Propagation" Algorithmus parallel implementiert. Das Programm beherrscht sowohl kontinuierliches, als auch nicht-kontinuierliches Lernen und kann bis zu 247 Millionen Gewichtsadjustierungen pro Sekunde durchführen. Damit ist es über 200 Mal schneller als moderne Tischrechner und sogar schneller als herkömmliche Höchstleistungsrechner. In einem nächsten Schritt wurde das Konzept für einen flexiblen und effizienten parallelen Simulator für neuronale Netze erarbeitet und realisiert. Mit einer einfach zu erlernenden Kommandosprache, dem Neuro-Basic, kann dabei der Simulationsablauf definiert werden. Den Abschluss der Dissertation bilden Ergebnisse von praktischen Experimenten mit neuronalen Netzen, welche mit dem Simulator durchgeführt worden sind, und es wird über erste Erfahrungen von Studenten im Umgang mit dem Simulator berichtet.

Abstract

In recent years artificial neural networks became very popular and of increased importance as an alternative approach to solving problems where no precise rules are known. In contrast to classical programming, neural networks are trained with examples of a particular problem in order to obtain a final solution. Research in this field requires the high experimental flexibility provided by simulating neural networks on digital computers. However, this simulation, and, in particular the learning process, requires immense processing power. So far computer performance was the limiting factor. The introduction of parallel computers offers new opportunities in neural networks technology.

This thesis contains contributions to the simulation of neural networks on parallel computers. For this purpose the MUSIC (Multiprocessor System with Intelligent Communication) parallel distributed-memory computer has been used. In a first step, the popular back-propagation algorithm was applied. Its implementation is flexible in the sense that it can simulate batch learning as well as continuous weight update and it computes up to 247 million connection updates per second. This is more than two orders of magnitude faster than can be attained with modern computer workstations and even outperforms present-day conventional supercomputers. In a second step the concept of a flexible and efficient parallel neural net simulator has been developed and put into operation. An easy-to-use command language called Neuro-Basic controls the simulation flow. To eliminate sequential overhead caused by communication latency between the host computer and the attached processor array, the command language is precompiled and executed exclusively on this processor array. The thesis concludes with some results of practical neural networks experiments utilizing Neuro-Basic and first experiences gained by students applying the system.